* + 1. Decide problem to solve
       1. Predict assessment performance based on gameplay data
    2. Come up with hypothesis and data to collect
       1. Hypothesis:
          1. Children that engage longer with the videos, games, and toys do better on the assessments. Look at:

Time spent on “Clips”

Time spent on “Games”

Time spent on toys? Don’t know how these are identified.

Correct responses on games

Exploration of toys, how to measure?

Number of objects interacted with – look for event

* + - * 1. Children that receive more instructions do better on assessment

Look for all events that contain ‘instruction’ in info column.

3110 – instructions to complete task

3010 – corrective instruction

3020 / 3021 – feedback on task

3120 – incorrect submission feedback

* + - * 1. Children who hit the help button more do better in the assessments:

Row 17 - event 93edfe2e

* + - * 1. Children with more dwell time with do better on assessments
        2. .
    1. Collect data
       1. Collected
    2. Clean Data
       1. Import data
       2. Preprocessing exploration

* + - * 1. Summary statistics
        2. Frequency counts for categorical data

Look for anomalies (1, 2, 3, “Fred”?)

Use .info and look for ‘object’ data types, this suggests mixed data types (numeric and string), indicating bad values in a column

Coerce values to the proper data type. Remove / replace / impute those incorrect values.

* + - * 1. Visualization

Histogram, PDF, CDF

Boxplots

Scatter plots

Swarm

Violin

* + - * 1. Tidying data

Melt

Pivot

Reset index

Split with .str

.split()

.get()

Detect missing values

* + - * 1. Combine data

Concatenation

Merging

1 to 1

Many to 1

Many to many

* + - * 1. Cleaning data

Convert data types

String parsing with regular expressions

Extract numerical values

Pattern matching

Functions to clean data

Lambda functions

Duplicate and missing data

Drop duplicate data

Fill missing data

Impute for missing values.

Test with asserts

Time Series data

Reindex the index

Resample time series data and chain methods

Rolling mean and frequency

Filter data

Interpolate missing values (.interpolate)

Set time zone and convert appropriately

* + 1. Join multiple data sets
       1. Merge train\_labels[‘accuracy group’] column to train dataset so that all event\_ids have an associated accuracy group
       2. Extract all installation\_ids that took assessments to train\_assess
       3. Merge specs[‘args’] to train\_assess
       4. Extract all events with dwell time events to train\_assess\_dwell
    2. Manipulating data frames
       1. Extract and transform the data
          1. Organize index and columns of dataframe
          2. Filter dataframe to extract interesting data
          3. Threshold data
          4. Transform dataframe

.apply() method

.map() method

Vectorized functions

* + - * 1. Advanced indexing

Set and change index

Build and manipulate hierarchical indices with Multiindexing

* + - * 1. Rearrange and reshape data

Pivot data

Stack / unstack data

Melt data

Use pivot tables

* + - * 1. Group data

Group strategies:

By column

By another series

By function of the index

Aggregate

On index levels / fields

Group and transform by group

Group and filter:

With .apply() using a function

With .filter() using a lambda function that generates a Boolean series

With .map() using a dictionary that maps the values of a series to another set of values

* + 1. Merge dataframes
       1. Merge specs with event\_code, one example event\_data entry and order by frequency to get idea of what each type of event does
       2. Merge train\_labels with train by ‘installation\_id’ to get the ‘accuracy\_group’ for each event in train
    2. Exploratory data analysis
       1. Formulate relevant questions, and hypotheses
          1. Children that engage longer with the videos, games, and toys do better on the assessments. Look at:

Time spent on “Clips”

Time spent on “Games”

Time spent on toys? Don’t know how these are identified.

Correct responses on games

Exploration of toys, how to measure?

Number of objects interacted with – look for event

* + - * 1. Children that receive more instructions do better on assessment

Look for all events that contain ‘instruction’ in info column.

* + - * 1. Children who hit the help button more do better in the assessments:

Row 17 - event 93edfe2e

* + - * 1. Users with longer dwell time will have higher accuracy
        2. Users with more dwell time events will have higher accuracy
      1. Test those questions with visualizations
         1. Covariance plot of accuracy group with counts of each variable # of clips, # of activites, # of games
         2. Plot total time spent in each world prior to assessment vs accuracy group
         3. All worlds

3D plot of # of games, clips, activities on xyz axis with accuracy group as the hue

* + - * 1. Plot data of mean dwell time vs assessment score / accuracy / accuracy group
      1. Identify trends in the data
      2. Look for relationships between variables
      3. Communicate results with visualizations (scatter plots, histograms, etc.)
         1. Seaborn plots

Simple Linear regressions

Residual plots of regressions

Higher order regressions

Group Linear regressions by hue

Univariate distributions-

Strip plots

Swarm plots

Violin plots

Boxplots

Multivariate distributions

Joint plot – visualize two distributions simultaneously

Types:

Scatter

Regression

Kernel density estimation

Hex bins

Residuals of a regression

Scatter\_matrix – similar to jointplot with scatter

Pairplots – plot joint plots for all of pairs of non-categorical variables in a dataframe

Can add a hue for a categorical

Covariance matrix – quickly gauge correlations between many variables

* + 1. Feature selection and engineering
       1. Dwell time
       2. Investigate what other features can be extracted from gameplay data
    2. Train models
       1. Supervised learning methods (since you’re given labeled data)
          1. Classification

K-NN

Handle missing data – impute or drop

Center and scale – normalize etc.

Split up into train, and test sets

Fit model

Tune hyperparameters

Evaluate model:

Cross validate

Accuracy

Classification report

Recall

Precision

F1 score

* + 1. Evaluate models
    2. Iterate and improve models
       1. Revisit feature selection and engineering
       2. Try new models
          1. Logistic regression

Same procedure as above.

Evaluate model

Receiver – Operator curve (ROC)

Area under the curve (AUC)

* + - 1. Model improvement methods: